Authorship Attribution and Profiling in Spanish and English Language

Master Thesis

University of Fribourg

Lesly Miculicich

September 10, 2014

Supervisor: Dr. Jacques Savoy, Computational Linguistics, Institut d'informatique (IIUN) Université de Neuchâtel

Abstract

The authorship attribution is the practice of inferring the author of a given text based on the analysis of her/his writing style. It has been largely used in literature work disputes but it has other interesting applications such as forensics and plagiarism detection. The purpose of this project is to experiment and present a solution that can identify the authors of a given corpora. We have two corpora to analyse: Spanish literature of the 19th century and blogs written in English and Spanish. We aim to identify the author given a list of candidates or infer its gender or age range. We propose to use the Kullback-Leibler Divergence (KLD), an information-based measure of disparity among models. In order to validate the proposal we use as baseline the naïve Bayes classifier whose performance is generally accepted for this kind of problem. The results show a significative improvement with the proposed method over the baseline when there is enough text size to train, and they were really promising when detecting the gender and age in the blogs in English language. The performance using few data training could improve with some input conditions identified and described in this report that could be a precedent for future work.

Keywords: Authorship attribution, authorship profiling, text classification, stylometry

Acknowledgements

I can attribute to myself have written this text, but for the rest thank you mother, you are my support in everything.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Motivation

Determining who was the author of a doubtful text can be useful in a wide variety of tasks, since solving disputing literature works until clarifying criminal cases. One popular example is the Shakespeare's authorship question, over the years, different specialists have questioned the origin of the books of this famous dramatist generating a long time discussion among experts. As a result Shakespeare's work was disputed among more than 70 different authors such as Francis Bacon or Christopher Marlowe. Within this framework, Craig and Kinney [9] performed a authorship attribution analysis comparing Shakespeare's and other writers' styles and their results confirmed in many cases the scholarly consensus.

Another example of application is when digital evidence is involved in a crime such as an email, a suicide note or an electric blog post. In these cases the question is *"Who was at the keyboard when the relevant documents were produced?"* [8], some criminal cases including homicide, racial intimidation and sexual exploitation of children are described by Chasky [8] and he proposed an interesting analysis to identify the authors of the evidence to help to clarify the cases.

Furthermore, over the internet are thousands or millions of content texts in blogs pots, forums, reviews, tweets, news, etc. This information is valuable for different purposes, for example, if there is a suspicion that any content is dangerous, reaching for the author can be assertive; that is the case in the analysis made by Abbasi and Chen [7] about web sites and forums known to be used by extremists groups . Even if identifying the specific name is not possible, knowing the age rage, gender or any other characteristic helps.

1.2 Problem Description

The authorship attribution can be defined as "The science of inferring characteristics of the author with characteristics of the documents written by the author" [21]. This problem has a long history, it dates back to 1851 [17] when the English logician Augustus de Morgan suggested to his friend Thomas Mendenhall that the problem of authorship may be deal in the length of words. Mendenhall then published their studies [16, 29] about Shakespeare, Marlowe and Bacon; however, it did not have a great impact on the literature experts of the time. It was years later, after the arrival of computing and modern statistics, when the *non-traditional* or statistical authorship attribution arose.

This practice is based on the *stylometry*, Holmes summarises this concept saying: "At its heart lies an assumption that authors have an unconscious aspect to their style, an aspect which cannot consciously be manipulated but which possesses features which are quantifiable and which may be distinctive." [17] those features are the author's *fingerprint*, and by measuring them the authorship attribution try to identify him. Thus, what exactly are the elements that compose the stylistic fingerprint? this is the question that scholars and practitioners have being trying to solve. For the moment, there are some clues: words, functional words, syntactic notation, semantic dependences, etc. that tend to work well with specific text corpora and problem type.

The three main classical problems [21] in authorship attribution are:

Closed class "Given a particular sample of text known to be by one of a set of authors, determine which one".

Open class "Given a particular sample of text known to be by one of a set of authors, determine which one, if any". In this case we may do not have previous data from the given author, which makes the problem harder to solve.

Profiling Determinate any property of the author given a sample text, it could be author's gender or age; weather he is native speaker or not; or if it is a case of multiple authorship.

However, more recently applications have raised some more problems [22]:

Needle-in-a-haystack Thousands of candidates and very little data sample per each one.

Verification There is not a closed candidate set but there is one suspect, determine whether or not it is the author.

Since the computer science perspective, there are three areas that are the base for authorship attribution technologies: *a*) Information Retrieval (IR), with techniques of document retrieving to measure the *distance* between the author's writing style (characteristic vocabulary) and the documents vocabulary; *b*) Machine Learning (ML), with classification techniques where the classes are the authors and the features are the author's writing style (words, n-grams, characters, etc); *c*) Natural Language Processing (NLP), with analysis of the text structure, grammar and semantic correlations in order to find new patrons in the author's style.

1.3 Challenges

Despite the efforts of years of research and development of technology, the authorship attribution practice has still challenges to surpass.

- The methods need a number of samples of text from the questioned author to analyse and extract the characteristic features of its writing. One of the main problems is that usually there are very few samples. With literature, we have a significative quantity of text and for that reason the methods have a very good prediction accuracy, nonetheless, when analysing e-mails, forums or tweets, the text length decrease drastically and the classical methods have low accuracy rates. Because of this, new studies are trying to recognize and incorporate more features that identify the writing style.

- Until the moment, it is not clear what are the factors that characterise the writing style. There are some clues that work in specific cases, but there is not yet generic and clear rules that demonstrate their universal validity. This is the main argument of forensic experts, linguists and scholars to accept this practice as sufficiently reliable.

- With more elaborated features such as semantic information, the language models have more errors. Due to the technical difficulties to understand automatically the natural language, it is not an easy task for computer to understand the meaning or even to capture the grammar, many noise is added to the models which derives in errors in the analysis. Therefore, more powerful techniques of natural language processing are needed.

- The independence of topic is also another challenge, the idea that the stylometry defends is that the author style is independent of the genre, topic, or literary style: e-mail, a novel, business report or a review about cars. With the current methods is difficult to separate what really represents the fingerprint and what is extra information.

- The understanding of the author profile is a relatively new challenge as well, how its gender, age, background, education level, native language or even psychological condition affects its writing. Some scholars have appealed to other fields such as linguistic or psychological searching for clues, maybe more interdisciplinary studies will be needed.

- One last aspect to mention is the language, the treatment of style features may vary depending of the language of the written text, Chinese writing is far away from English. However, the idea of finding the author's fingerprint remains independently of the language. In addition, not many studies have been done in other languages besides English, consequently, there is an empty niche for research.

1.4 Project Purpose

The purpose of this project is to experiment a solution for two types of problems in the authorship attribution framework. The first one is the *closed class* problem, identify the author given a set of possible candidates; and the second one is the *profiling* problem, identify the gender and age range of the author.

The challenge is to work with different types of text content, the corpora consist on literature work, representing the case when we have enough text to train; and blogs, representing the cases when there are relative short text; in Spanish and English language.

The idea is to compare the the proposed solution with a baseline and understand which are the relevant factors that helps to improve the accuracy in each case. With this work, we hope to contribute a little in the understanding of the writing personal style and increase the research over Spanish language.

1.5 Report Structure

The report is structured as follows: The first chapter is the introduction, it starts describing the motivation for this project following by the problem statement, a review of the current challenges and finally the purpose of this thesis. The second chapter presents the background and state of the art of the authorship attribution, with a special emphasis in the milestones in the history of stylometry. The third chapter is the description of the solution, we explain the methodology used, the formal approach and the implementation details. The fourth chapter is the evaluation and analysis of results where we compared the solution with the baseline and describe the obtained results identifying the contribution, scope and limitations of the solution. Finally, the conclusion of the work and the reference material is given.

Chapter 2

Background and State of the Art

Contents

2.1 Background and History

The oldest study in statistical authorship attribution dates from 1887 [16], when Mendenhall presented his analysis about word length frequency of writers [28]. In a following work [29] he found a especial patron in the Shakespeare's writing: he used significantly more four-letter words than three-letter words, which was the standard in all other studied authors, only Marlowe coincided with this great writer, that situation caused some controversy at the time. Since then, some other statistics such as word length, syllables per word and distribution of part of speech were proposed, nonetheless, they are not longer used because experts demonstrate their unreliability as accurate measures [21].

It was in 1964 when Mosteller and Wallace and their analysis of the Federalist Papers [30] demonstrated to be accurate enough to represent an alternative to traditional human-expert based techniques [21]. The Federalist Papers [18] are a set of 85 essays published under the pseudonym of Publius between 1787 and 1788. They supported the ratification of the new Constitution of the United States. By the time it was known that the real authors where John Jay with 5 essays, Alexander Hamilton with 14, and James Madison with 54; however, 12 essays were disputed between Hamilton and Madison. Mosteller and Wallace where the first to apply probabilistic and computational assistance to solve this problem, they applied a Bayesian analysis over a set of words. They tested different words between *function words* and *content words* an proposed a list that better discriminate the author's style. The Federalist Papers have became a standard corpora to test new methods because of the document availability, well defined authorship candidates and homogeneity of topic, genre and literary style [21]. The seminal made by Mosteller and Wallace was the baseline reference for following works.

Until the 1990s, hundreds of studies have being done. We can mention the study of Zipf [45] about the word frequencies where he observed that in a writing text few words were highly frequent while most of them depict a low frequency; the result was the *Zipf Law* (the frequency is inversely proportional to its frequency rank). Additionally, Discriminant Analysis and Clustering Analysis were used, they allow to observe which variables are more suitable to discriminate authors. Holmes [17] presents a more extended review over this part of the history. The new methods based on computer process, statistics and stylometry encourage to practitioners from different areas outside the traditional ones (literature experts or forensics) to make their own analysis of writers. However, the non-expertise over the written topic and the lack of formalism in the methodologies [34] caused suspicions regardless the credibility of the results. There were some famous controversies: the *Cumsum* and the *Elegy* cases [21]. The *Cumsum* is a technique proposed by Andrew Q. Morton that was adopted by English courts in 1991 and 1992; though, it was immediately criticized for the accuracy issues and gained mistrust from scientific community. The work of Don Foster suffered the same fate when he claims that the poem *"A funeral Elegy"* was wrote by Shakespeare, and afterword the scholar consensus was in favour of John Ford's authorship. The problem with those early methods was the lack of objective evaluation [36]: the texts were too long, testing data were not controlled by topic, there was an intuitive evaluation and the comparison between methods was difficult. By then, the *non-traditional* authorship attribution was severely questioned even Rudman in 1997 [34] maintained that "Results of most non-traditional authorship attribution studies are not universally accepted as definitive" clamming that the studies were governed by expediency, lack of competent research, corrupted data among other problems.

Nevertheless, the authorship attribution has change the last years [36]. It was Burrows [7] [6] who regenerates the stylometry as a viable tool [17], he applied Principal

Component Analysis (PCA) over a set of functions words and plotted the two first components. The results showed a very clear clustering of points representing each author's text, "the data speak for itself" [17]. This successful and a number of additional events marked the turning point. The large quantity of data over the internet; the evolution of the techniques of NLP, ML and IR; the new applications in literature, criminology, plagiarism, marketing and others have yielded improvements in the methods. Additionally, scholars analysed and prosed solutions for the problems, for example Rudman [34] suggested to have a complete and correct experimental design, education to practitioners and a deeper study of the style and Juola [21] suggested to improve the evaluation and proposed a method for comparison. Finally there were some studies that help to improve the general accuracy [36]. As a result, we have now a promising research area with a significant advance in the methods and technology, objective evaluation criteria, good comparison methods and standardised text corpora.

2.2 State of the Art

This section presents a summary of the current methods, technologies, practices in authorship attribution and their theoretical support. A typical structure of the the process is plotted in the Figure 2.1, next, each part of this process is described.

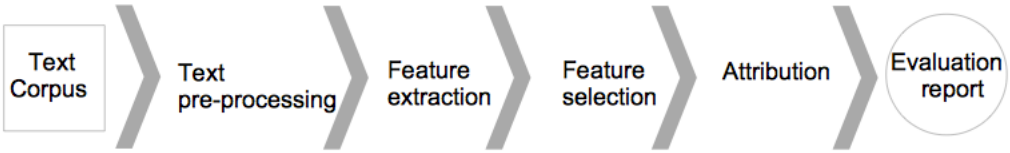

Figure 2.1: Typical process of the methods applied for authorship attribution

2.2.1 Text Corpora

In order to validate the results and compare the performance of the authorship attribution methods, different text corpora data are available. One of the most used over the time is the collection of Federalist Papers accessible in a number of web sites such as the Project Gutenberg [24], this web site also has a collection 45,000 ebooks in 55 languages, a variety of literature works can be obtained there. Some researchers use corpora from online resources such as newspapers, forums, tweets, blogs and so on. In [36] we can find a extended revision of them. Corpora created for other purposes is also adapted, for example the TREC or Brown [12] collections, and many others in different institutions web sites: Stanford NLP Group [14], the National Centre for Text Mining (NaCTeM) [11] and PAN [33].

A good corpora for testing [21] should be homogeneous in the topic, genre and literary style, and written in the same period of time. It is important to notice that the data must be separated at least in two disjoint sets: training and testing. In some cases one more set is added for validation. The corpora selection depend on many factors: problem type, corpora total size, distribution per author and some other aspects. The most import evaluation parameters for corpora [36] are:

- Training corpora size

- Test corpora size
- Number of candidate authors
- Distribution of the training corpora over the authors

2.2.2 Text Pre-Processing

This first task, not always described, is the pre-processing. Usually the given text corpora contains *noise*: metadata, format characters, structure information (number of page, footnotes, index, etc.). We need to clean and extract the relevant text. Furthermore, text from different sources has different structure, for example the literature may be in simple text format while forums text may be in XML format. It is helpful to convert them into an standard format [38]. The result would be a pure text and its relevant information, such as author or gender labels, in a format that the program can read.

2.2.3 Feature Extraction

One, if not the most, difficult task is to recognise the writing style or the stylistic fingerprint; this set of features exists as a unique set of characteristics that varies from one person to anther. This assumption facilitates their identification regardless of the content topic, genre, the type of writing (formal or informal) and even whether they are aware or not conscious that they can be identified. This second task consist in extracting these features from the text and representing them in a model.

2.2.3.1 Linguistic Features

The most relevant linguistic features that support the current studies in stylometry are:

Specific vocabulary differences and misspelling

One first idea is to see the use of vocabulary [21]. Its clearly to see the differences of the same language depending on the region and period of time. For example between British and American English: *"colour"* and *"color"*; or in archaic English: *"coz"* for *"cousin"*. Another clue is the misspelling, typically people use to repeat the same mistakes. However, it could be useless to base the analysis only in these words because, statistically, the quality of differentiate words is small and they could not appear in the given text.

Vocabulary Richness

The range of vocabulary among people can vary drastically. For example a native English speaker will probably use more different words than a non-native speaker. Technically, it is the quantity of different words used by the author over the total number of words. Unfortunately the measure is not really accurate [21]

Word Frequency

Although the previous features can not be effectively applied alone, one way to leverage its advantages is by calculating the frequency of use of each word by the author, which gives an idea about how he use the vocabulary. It is a simple but sufficiently effective solution. The disadvantage is that it dismiss information about the order of the words and the context. [36]

Punctuation

It also, as the same manner as vocabulary, can be another type of measure [21]. This type of feature can be important to analyse, for example, when the data comes from blogs, tweets or forums where people tend to write emoticons.

Function Words

In traditional grammar the word classes (or parts of speech) are divided in *function words* and *content words* (or lexical words). The *function words* (or grammatical words) [1] are words that have grammatical importance rather than lexical meaning. Their function is to merge or connect *content words*. They are: auxiliary verbs, determinants, articles, conjunctions, pronouns, prepositions and modal verbs.

In English language and in many other languages, the *function words* mostly correspond to the more frequent written words [21]. In the Figure 2.2, we can see the table that Joula made over the Brown corpora where roughly they are listed first in the ranking.

High frequency		Medium frequency		Low frequency	
Rank	Type	Rank	Type	Rank	Type
1	the	2496	confused	39996	farnworth
2	of	2497	collected	39997	farnum
3	and	2498	climbed	39998	farneses
4	to	2499	changing	39999	farmwife
5	a	2500	burden	40000	farmlands
6	in	2501	asia	40001	farmland
7	that	2502	arranged	40002	farmington
8	is	2503	answers	40003	farmhouses
9	was	2504	amounts	40004	farmer-type
10	he	2505	admitted	40005	farmer-in-the-dell

Figure 2.2: Samples of high frequency, medium frequency, and low frequency words from the Brown corpora (taken from [21])

This class of words lack of lexical meaning and tend to be topic-independent, for these reasons, they perform well for authorship attribution [21]. They are part of the stylistic fingerprint because their use varies from one person to another, sentences written by two different authors can have the same meaning but different grammatical constructions. Mosteller and Wallace [30] were told by the scholar Douglass Adair that the words *"while"* and *"whilst"* were good discriminators between Hamilton and Madison therefore they made their own experiment with *"by"*, *"from"*and *"to"* obtaining as a result highest rates of use of *"by"* for Madison and *"to"* for Hamilton.

Marked Words

The markedness [1] is applied to many areas of language. In the Table 2.1 we can see some examples.

This also is a discriminator among authors for example one person would never use passive voice while other could rather use it frequently [21].

Table 2.1: Examples of Marked Words

	Unmarked	Marked
Sentence	*"I love Lucy"*	For negation: *"I don't love Lucy"* For interrogation: *"Do I love Lucy?"*
Nouns and verbs	*"Look"* *"Table"* *"Nice"*	For past : *"Looked"* For plural : *"Tables"* For comparative : *"Nicer"*
Semantics	*"Horse"*	For sex: *"Stallion"* and *"Mare"*

Syntactic and Part of Speech

The authors tend to unconsciously use similar syntactic patters. Therefore syntactic information may be consider a more reliable fingerprint than lexical information [36]. The problem is that this measure depends on how well we can recognise automatically the natural language and usually the process presents errors [21]. One way to annotate syntax is by detecting the *part-of-screech (POS)*, for example, the Stanford Part-Of-Speech Tagger [37] analyses the sentence *"A passenger plane has crashed"* as following:

A-DT passenger-NN plane-NN has-VBZ crashed-VBN

Where each word of the sentence is *tagged* with the syntactical class: determiner (DT), verb (VBZ, VBN) and noun (NN). Another example is by *chunking* the sentence in higher structures such as noun phrases (NP), verb phrases (VP) or prepositional phrases (PP) [36]:

NP[A passenger plane] VP[has crashed]

N-gram

A *n-gram* is a sequence of n consecutive parts of the text: characters, words, syllables, etc. For example, if we use a *two-gram* analysis for the sentence *"I love to dance"* the result is:

"I love", "love to", "to dance"

This is another way of add contextual information, in the previous example the word *"dance"* can change meaning depending on the previous word. For example *"to dance"* is different from *"a dance"* [21]. The problem with *n-grams* is that they add to much dimensionality which have a bad effect for efficiency.

Semantic Properties

Textual meaning also represent the author's *fingerprint*, aspects such as feeling, genre and personality are differentiators of the text style. For example according to the study [3] females tend to write more negative feelings and males positive ones. Therefore not only the syntax or grammar feature are important but also the semantic ones. One way to incorporate semantic information is by annotating semantic dependencies, for example the Stanford Dependencies parser [10] would analyse the sentence *"Bell, based in Los Angeles, makes and distributes electronic, computer and building products"* as showed in Figure 2.3.

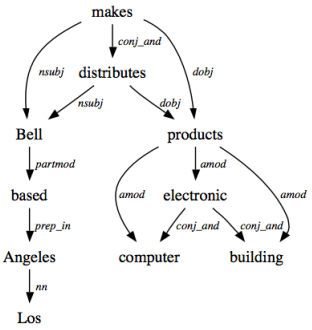

Figure 2.3: Example of semantic dependencies notation (taken from[10])

In this case the sentence is represented in a graph were the nodes are words and the edge labels are grammatical relations.

Orthographic Properties (Characters)

There are also morphologically related words that could be part of the *fingerprint* style. The idea is that a person who writes *"dance"* is also likely to write *"dancing"*, *"dancer"*, etc. Then *"danc"* could be a characteristic feature. In order to take advantage of this, practitioners use characters sequences [21]. An additional approach is to take *n-gram* of characters [36] to add some contextual information, for example in the text *"to dance"*, a *four-gram* analysis would take *"to d"*. This approach is also tolerant to noise, as for example with the words *"simplistic"* and *"simplistc"* would take similarly *n-grams*. The application of this type of analysis has prove to be useful specially with short texts such as e-mails or tweets [36].

2.2.3.2 Language and Features Model

In order to implement a solution, we need models for representing the the text and the linguistic features. Some examples of language models used in Natural Language Processing are *context-free-grammar* and *bag-of-words*; and for features, the model most commonly used is the *vector space model*.

Bag-of-Words

The *bag-of-words* model is wide used in Natural Language Processing. It ignores the structure and linear ordering of words and just represent the text as a list of the words that compose it. Consequently, the context is dismissed assuming that the words order is irrelevant [26]. The problem is the highly correlation among the words [21]. For example documents written in first person will have hight frequency of words as *"I"*, *"me"* or *"am"*. However, it has prove statistically be a good enough model. For example, the sentence *"The velocity of the seismic waves rises to the same velocity as those of the previous earthquake."* represented in the *bag-of-words* model would be:

[*"the"*, *"velocity"*, *"of"*, *"seismic"*, *"waves"*, *"rises"*, *"to"*, *"same"*, *"as"*, *"those"*, *"previous"*, *"earthquake"*]

Context-Free-Grammar

The *context-free-grammar* is a model that aims to isolated interpret the syntactical structures by recognising the categories and function of the words [26]. The advantage of this model is that includes contextual information. The problem is that they are highly susceptible to errors of interpretation. For the before example the *context-free-grammar* model is showed in Figure2.4.

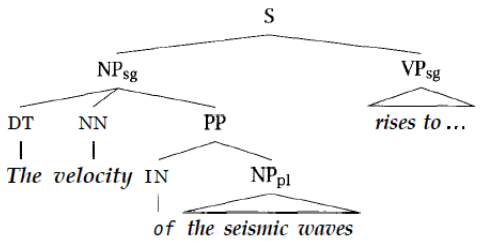

Figure 2.4: Example of *context-free-grammar* model (taken from [26])

Vector space model

The features are typically represented in a vector space. For example, if we use a *bag-of-words* language model, each dimension of the vector would be one word and a document will be represented as a vector of words [21]. One problem with this representation is that is highly dimensional, corresponding to the length of the total corpora vocabulary.

2.2.4 Feature Selection

In authorship attribution, as we see before, a big quantity of features need to be managed. Some of them are just *noise* that will not help in the discrimination process. Therefore a reduction or selection of features sometimes is fundamental for the performance.

2.2.4.1 Dimensional Reduction

One of the first attempts to reduce the dimensionality was applying Principal Component Analysis (PCA) [21]. In this case, the method manipulates the vector space and tries to visualize the data by simply capturing most variance. It reduce a vector set of correlated values in a smaller set of uncorrelated ones. The problem is that this method is not well optimised for classification tasks because the most variables features not always are the ones that represent the writing style.

2.2.4.2 Data Exploring

It is useful to explore our data to understand it. This understanding will help to make a more precise idea about what features are relevant and what methods we can apply for attribution. Some methods that do not require information a priory and allow to make a first plot of the data are Principal Component Analysis (PCA), Multidimensional Scaling and Clusters Analysis [21].

2.2.4.3 Feature Selection

Some studies focus on the discriminatory power to automatically recognise the optimal features given a training dataset of documents. For example applying Genetic Algorithms [36], Information Gain [21], or just by selecting the more frequent features [36]; nevertheless, the closeness with the training data can make them lose generality and do not work at the same way with another set of data so we should be careful.

Finally, the output of this part of the process should be a list or a vector of features labeled, depending on the problem, by author, gender, age range and so on. They correspond to the data set for training (known authorship) or testing (unknown authorship).

2.2.5 Attribution

There are two different treatments for the training set: individually per document or cumulative per author. Stamatatos [36] called them: *instance-based approaches* and *profile-based approaches* respectively.

2.2.5.1 Profile-Based Approaches

These approaches cumulate or summarise all documents by their corresponding author. Consequently, one single item (vector) represents the author's style. Each testing document is compared with this profile vector. In the Figure 2.5 we can see the architecture of this methods.

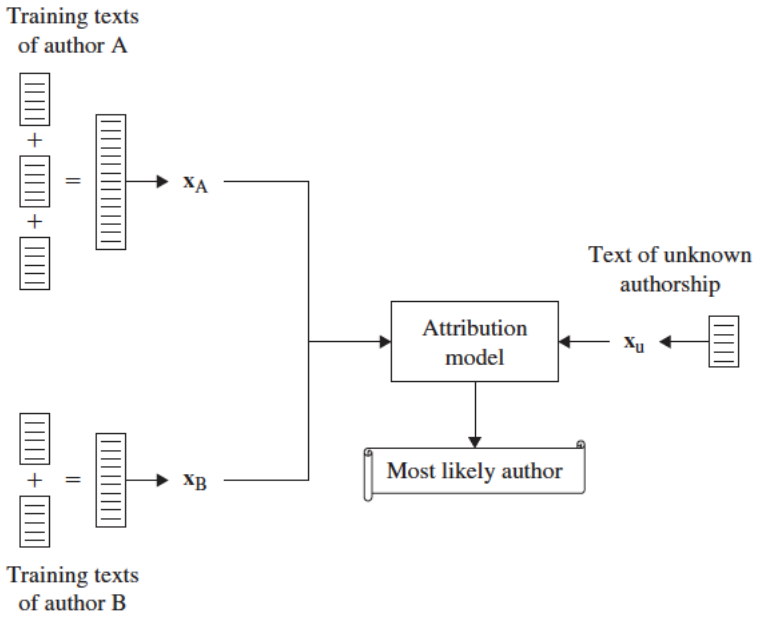

Figure 2.5: Typical architecture of profile-based approaches. "Note that x denotes a vector of text representation features. Hence, xA is the profile of Author A, and xu is the profile of the unseen text." (taken from [36])

This methods do not have a special training process, it just corresponds to the cumulative feature extraction. Then, typically they measure the distance between profile vector and the feature vector of the unknown document, what vary in each method is the

distance function. Finally the attribution of the class correspond to the minimum distant candidate author.

Stamatatos [36] classified those methods as following:

Probabilistic models These methods attempt to maximise the probability that a document x belong to an author a, $P(x—a)$. They have archived a hight performance in the experiments in comparison with other techniques, and they obtain better results using word-based models.

Compression models These methods do not extract directly one single vector of features but concatenate all feature-vectors of documents from a given author and compress them in one single file. Similar method used by RAR or ZIP. One problem with this methods is that they can not be applied to all features types.

Common *n-grams* and variants These methods concatenated all documents of an author an extract the most frequent *n-grams*. They perform well with balanced training set (almost the same quantity of document per author) but not so well with unbalanced ones. One strategy is to try to balance the training taking only the same quantity of documents per author.

2.2.5.2 Instance-Based Approaches

These methods are Machine Learning based, therefore, they require a previous training process before classify or attribute the authorship. In Figure 2.6 we can see the architecture.

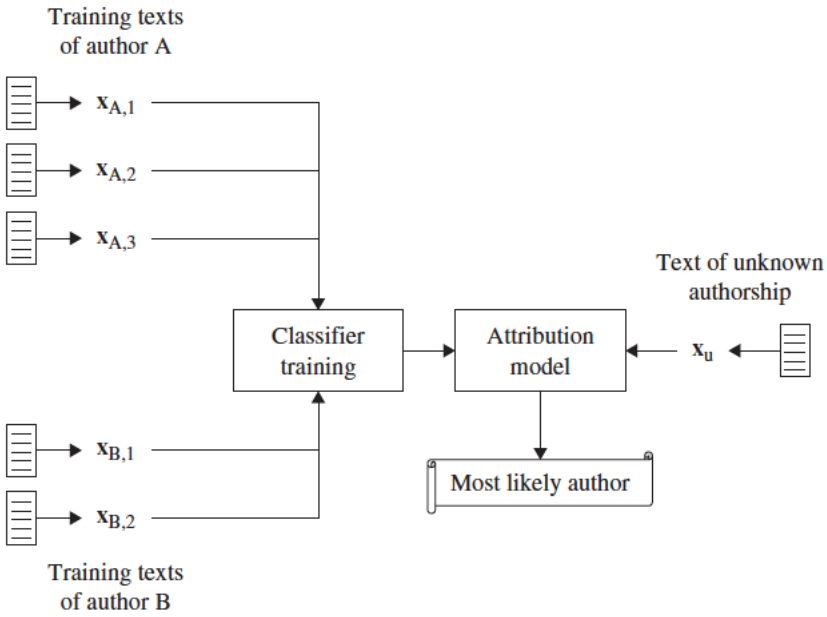

Figure 2.6: Typical architecture of instance-based approaches (taken from [36])

Vector Space These methods includes: Discriminate Analysis, Support Vector Machine (SVM), Genetic Algorithms and some others. They usually do not need a feature reduction or selection because they can manage a high diminutional space representations. Moreover, they can manage unbalanced training data.

Similarity-Based These methods calculate the similarity in pairs and use the nearest neighbour algorithm to attribute the author. Some examples are: Delta (based on z-score) [36], Kullback–Leibler divergence (KLD) and Kolmogorov–Smirnoff distance (KSD) [21]. The results vary according the method used but in general they have a good accuracy with large set of training documents.

Machine Learning Classifiers A number of classifiers from Text Categorisation problems are used with good results for authorship attribution problem. Here we describe some of the most applied.

Decision trees They are designed to support descriptive classification and clearly explain the reason of their decisions. However, they could have less performance than other methods [21]

Naïve Bayes classifier It is based on Bayes theorem to infer a classification. It is naïve because it assumes independence of the variables. It can performs very well and is easy to implement and train [21].

Neural networks Usually they produce very accurate results; however, it is not really clear in what basis they make their decisions [21].

2.2.6 Evaluation

Evaluation is the key of the development of the authorship attribution. In real cases we only have the training set, nevertheless, the performance of the training set is not always the same when we use a real test set. When we have a large amount of data training is not so much problem because statistically will resemble the general cases [39]. The problem is when we have small amount of data training, in this case we need to measure the cost of the error rate (Is it acceptable enough for this problem?). There are standard evaluation measures that help to validate the solution and compare it with other methods.

2.2.6.1 Training and Testing

As mentioned before, the performance of the training set almost never resembles the real performance of the solution. For this reason, it is needed a separate testing set. Both, training and testing sets, must be representative sample of the problem. Sometimes there is a third set called validation set which is used to optimise the parameters of the method after training, and the testing data is used to obtain the evaluation measures.

2.2.6.2 Cross-Validation

It is one of the most simple and used methods for validation. Ideally, we have enough data to separate a training dataset and a testing dataset, however, this is not always possible. In theses cases, cross-validation split the training and use one part for testing. The are different ways to split the data but *k-folds* is the most used; and particularly, *10-folds* is a reference to compare different approaches. In the *k-fold*, the data is split in k equal parts, one of them for testing and the rest for training. This process is repeated k times by alternating each part as the test part. Then, calculated the *loss function* for each fold and for the total *k-folds*. [15]

2.2.6.3 Student T-Test

In order to compare different methods and to define if one is significative better to another, we can use a statistical equivalence test. Given two samples of n results of two systems, we measure if there is a significative difference between them. The *T-Test* use a null hypothesis of equivalence, for example, assuming that both systems have an equal performance. Then, it measures the difference to accept or reject the null hypothesis. If the hypothesis is rejected we can say that there is an statistical difference. (more detail in [39])

2.2.6.4 Evaluation measures

There are different measures to evaluate the methods. In authorship attribution studies is commonly to use the *accuracy* and the *precision*. (more detail in [39])

Accuracy It can be defined as the measure of closeness to the real value. In this case is the number of documents well classified over the total number of documents.

Precision It is measure related to the capacity of the methods to reproduce or repeat its results. In this case is the number of documents well classified from one author over the number of documents classified to be of that author.

Chapter 3

Solution

Contents

3.1 Methodology

This project is a experiment with authorship attribution methods in order to measure and evaluate their performance with a specific defined corpora of text. For that purpose, the methodology used is as follows:

- The first step is to propose a initial solution as a baseline for comparison, this initial solution will be selected from the already existing methods, with the criteria that it must be simple and must has demonstrated its validity to solve the problem.
- The second step is to make a research over the exiting studies and propose the solution that better fits with the problem. Explaining the reasons for the decision.
- The third step is to implement the solution. It is needed to select the most convenient technological tools and programming language.
- The fourth step is to evaluate and analyse the solution results to verified if there is improvement respect to the baseline.

3.2 Baseline

The baseline will be use to compare the proposed solution. As mentioned before, the baseline for this study must be simple to implement in order to not dismiss time resources in a difficult implementation. Nonetheless, it must work sufficiently well in the context of the problem in order to represent an alternative solution. The naïve Bayes classifier complies well with those conditions. It is simple but effective in a number of classification problems [39] including the authorship attribution problem [21, 31, 32]. Moreover, this method is commonly referenced as baseline in a number of studies [41]. In addition, there are a quite number of libraries for its implementation available in different languages programs.

The naïve Bayes classifier[39] is a supervised classifier that assumes independency among the features. Given a class variable y and a set of features x_1 to x_n we have the following formulation:

$$P(y|x_1, ..., x_n) = \frac{\mathrm{P}(y)P(x_1, ..., x_n|y)}{\mathrm{P}(x_1, ..., x_n)}$$

With the assumption of independence, it is transformed to:

$$P(x_i|y, x_1, ..., x_{i-1}, x_{i+1}, ..., x_n) = P(x_i|y)$$

For all i it becomes:

$$P(y|x_1, ..., x_n) = \frac{\mathrm{P}(y)\prod_{x=1}^{n} P(x_i|y)}{\mathrm{P}(x_1, ..., x_n)}$$

Sine $P(x_1, ..., x_n)$ is constant, the commonly classifiers the following simplification:

$$P(y|x_1, ..., x_n) \propto P(y)\prod_{x=1}^{n} P(x_i|y)$$

Finally to obtain the class, they choose the most probable one:

$$\hat{y} = \arg\max_y P(y) \prod_{x=1}^{n} P(x_i|y)$$

We are using the *NaiveBayesClassifier* from the NLTK library [4].

3.3 Proposed Solution

In this section we present our proposed solution. First, we introduce the proposal and its support. Then, we present the detailed description and theoretical basis; follows by the advantages of the solution. Finally the implementation tools.

3.3.1 Presentation and Support

In general terms, the current attributions methods perform well enough with determinate conditions: problem type and corpora characteristics. There have being studies that aimed to compare them and give a more precise information for decision taking; some of them compare the classification techniques [35, 19, 23], others the features [41, 23, 13, 2] and even some of them the corpora size [25]. Among all features, the ones who have obtained more acknowledge are: *function words*, *POS*, *n-gram* and *most frequent words*; and among the attributed techniques: SVM and Bayes-based approaches.

In particular, the study of Savoy [35] is a good reference of comparison because he uses the Federalist Papers set, which, as described before, is a very suitable corpora for benchmarking in this topic. The study compares the methods: Delta, Chi-square, KLD, Z-score, Nave Bayes and SVM, what is interesting to note is that classical methods as KLD and Z-score had the best performance. Therefore, based on this study, the proposed solution is to experiment with the KLD method, specifically with the one proposed by Zhao and Zobel [42, 43] because it was the one which was evaluated in [35] and the corpora data is similar to the managed by this project.

3.3.2 Detailed Description

For the development of the solution, we follow the process scheme that we synthesised after the analysis of the state of the art and presented in the Chapter 2.

3.3.2.1 Text Corpora

The two set of corpora managed in this project are: *a*) Literature work in Spanish and *b*) Blogs in English and Spanish.

a) Literature Work The selected literature work corresponds to 10 Spanish authors catalogued within the Realism Movement during the second half of the 19th century known as *Literatura española del Realismo (Spanish Realist Literature)*. The corpora is composed by 78 e-books and the authors are two females and eight males. The e-books were obtained from Project Gutenberg [24]. The length of many of these books easily surpassed the 500 000 words, therefore, they were divided into smaller files. The division was made in two different ways: one taking parts of 10000 lines and the other with parts of 3000 lines (each line of around 18 words). The division was made in order to improve the performance speed and to increase the training document set which helps in the cases of authors with few samples such as females.

We use a maximum of four random parts of each book in order to prevent that the author features were too redundant. The Tables 3.1 and 3.4 show the documents distribution.

Table 3.1: Distribution of the corpora Gutenberg by author

Name	ID	# Books	# Parts (3000 lines)	# Parts (10000 lines)
Armando Palacio Valdés	ARMA	16	24	23
Concha Espina	CONC	3	3	3
Emilia Pardo Bazán	EMIL	4	5	4
Vicente Blasco Ibáñez	VICE	19	35	31
Benito Pérez Galdós	BENI	14	26	21
José María de Pereda	JOSE	5	9	9
Juan Valera	JUAN	10	10	10
Pedro Antonio de Alarcón	PEDR	4	5	4
Leopoldo Alas	LEOP	2	6	2
Fernán Caballero	FERN	1	2	1
Total		78	125	113

Table 3.2: Distribution of the corpora Gutenberg by gender

Gender	ID	# Books	# Parts (3000 lines)	# Parts (10000 lines)
Female	F	7	8	7
Male	M	71	117	106
Total		78	125	113

b) Blogs

This collection contains 235 texts from the PAN 2014 training corpora [33] for authorship profilin, in English and Spanish languaje. The files are in XML format distributed as showed in Tables 3.3 and 3.4

Table 3.3: Distribution of the corpora PAN 2014 by gender

Gender	ID	English	Spanish
Female	F	73	44
Male	M	74	46
Total		147	88

3.3.2.2 Text Pre-Processing

In this section, we describe shortly the pre-processing of the corpora, for this purpose we developed a program whose input are raw text files and the output are documents in standard format with only relevant text and labeled by author's name, gender and age.

a) Cleaning: For the Gutenberg corpora, the program deletes all the metadata and divides the files in equal size parts. For the PAN 2014 corpora, the program extracts the text from the files in XML and HTML format. The files were coded using the *universal character set UTF-8* to can managed special spanish characters.

Table 3.4: Distribution of the corpora PAN 2014 by age

Age	ID	English	Spanish
18-24	1834	6	4
25-34	2534	60	26
35-49	3549	54	42
50-64	5064	23	12
65-XX	65XX	4	4
Total		147	88

b) Standardisation: The raw files were in Text (with extension `.txt`) and XML format. We standardised them to a unique Text format (with extension `.txt`) . Each output document contains only text from one author and its labels are in the document's name in the following format:

`D[doc ID]_F[file ID]_A[author name]_G[gender]_R[age range]_L[language]_[n of file part].txt`[1]

Additionally, the documents were separated randomly in three groups: training (70% of the total), testing (20% of the total) and validation (the rest 10%).

3.3.2.3 Feature Extraction

As mentioned before, the features that seems to work better for authorship attribution problem are: *function words*, *POS*, *n-gram* and *most frequent words*. For the literature case we think that the *function words* or *POS* can represent well the authors' style; the *function words* can filter irrelevant information from the content such as places, characters, etc.; and, because the structure in this cases is regular and well developed, the *POS* can be recognized with few errors. For the blogs, we think that the function words can be a good discriminative feature. Therefore, we will use both of them. Addional they performed well with the KLD in previous studies [42, 43].

a) Language and Feature Model: The first step to extract our features is to decide how to represent the text. This study use the models described in Section 2.2.3.2. For language representation, we use *bag-of-words* and for the features representation we use *vector space* model.

b) Tokenization: The following step is the tokenization. It is the process to separated the text in parts or units called tokens [26]. Those tokens can be words, character, punctuation marks, n-grams or something else. Our tokens are words, however, we think pertinent to add punctuation marks because it can be relevant information in the case of blogs. For this task, we use the *word_tokenizer* of the NLTK Toolkit [4][5] which, for example, separates the sentence *"Good muffins cost $3.88\nin New York. Please buy me two of them.\n\nThanks."* as follow:

[*"Good", "muffins", "cost", "$", "3.88", "in","New", "York", ".","Please", "buy", "me", "two", "of", "them", ".", "Thanks", "."*]

c) Features: Finally, the features are extracted as follow:

[1]Where: `doc ID`: document ID (after processing), `file ID`: file ID (before processing), `author name`: four initial characters, `gender`: "F" or "M", `age range`: four length number (eg. 18-24 is 1824), `language` :"EN" or s"ES", n of file part: file partition number (if applies). When non-data "NN"

Functions words In this case, the list of tokens is filtered obtaining only the *function words*. For English, we used the same list as Zhao and Zobel [42] with 344 function words; and for Spanish we are using a list of 307 words. Following the previous example, we get something similar to:
[*"in", "me", "of", "them"*]

POS In this case, the text is tagged with syntactical information. For this task, we used the NLTK Toolkit [4] which has different options: *uni-gram, bi-gram, tri-gram,* or *hidden Markov* tagger. The *uni-gram* tagger applies its statistical algorithm word by word, the other taggers take *n-grams* to add a little contextual meaning. In the validation or parameter checking, we saw convenient to test the *uni-gram* and the *bi-gram*, all the rest had not significant performance difference. The tagger requires a previous tagged corpora to train. The corpora used for English was the Brown collection [12]; and for Spanish was IULA Spanish LSP Treebank [27], a tree-model tagged corpora based on 60,000 sentences annotated sentences from diverse topics. Following the example, the transformation would be:
"NP" - [*"ADJ", "NN"*] (in the case of "Good muffins")
"VP" - [*"VB", None, "NUM", ...*] (in the case of "cost $ 3.88 ...")

POS and Words Additionally to the tags we adjunct the words to add context and vocabulary information. Thus, we can differenciate words according to their function. We used the same tools as for extracting the POS tags. Following the example, in this case the transformation would be:
"NP" - [*"Good ADJ", "muffins NN"*]
"VP" - [*"cost VB", "$ None", "3.88 NUM", ...*]

3.3.2.4 Attribution

As already mentioned, the proposed solution is based on the previous work of Zhao and Zobel in [42, 43, 44], where they proposed to use Kullback-Leibler divergence (KLD). The KLD is widely used to retrieve documents, for example when it is required to search documents of certain author. Zhao and Zobel proposed an adaptation for authorship attribution problem, they use *function words* and *POS* in different corpora: English literature from Project Gutenberg, articles on different topics from TREC and some others. In [44], they tested the method and compare it with Naive Bayes and SVM obtaining a considerable improvement comparing to Naïve Bayes and similar to SVM. In [43], they tested the method with *function words* and *POS* with a relatively small collection (around 600 documents) with an *profile-based approach*. Finally in [42], they used a large collection of 500 000 documents, therefore, they changed their *profile-based approach* to an *instance-based approach* (it would be unpractical create a profile with hundreds of files). Then, they measured the distance of pairs of documents and applied a voting selection among the nearest 10 documents.

The Kullback-Leibler Divergence (KLD) [20] is an information-based measure of disparity among probabilities distribution. Although it is a positive measure, it is not formally a mathematical distance because is not symmetric. For that reason, it is called divergence. This measure is closely related to the concept of mutual mnformation and Maximum Likehood MLE, the MLE minimizes Kullback-Leibler divergence. The statement applied in information retrieval is:

$$KDL(d||q) = \sum_{x \in X} p_d(x) \log_2 \frac{\mathrm{p}_d(x)}{\mathrm{p}_q(x)}$$

Where $p_i(x), i : d, q$ is the probability mass function of getting the instance x. In a *profile-based approach* d is the model of the unknown document and q is the model for the author's profile, whereas in a *instance-based approach* q is the model for the know document. Normally, following a Maximum Likehood model we have the probability function: $p(x) = f_{x,d}/|d|$, where $f_{x,d}$ is the frequency of x in the document d and $|d|$ is the length of d. However, there is a problem when $p_i(x)$ is zero, for that reason, it is necessary to apply an smooth function. The Dirichlet prior is an effecting smoothing for text [40], it is given by:

$$p'_d(x) = \frac{|d|}{\mu + |d|} p(x) + \frac{\mu}{\mu + |d|} p_B(x)$$

Where, $p_B(x)$ is the probably of getting x in the *background* model. This model is a collection of prior probabilities, it can be obtained from an external corpora statistically representative. The idea is to bias the KLD to favour less common features. The parameter μ weight both models.

The background model were obtained from the Brown collection for English and from the IULA collection for Spanish. However, it is not always possible to obtain a good model that represents the general use of the language. For example we could not process a good model for the case of *POS* extracted with the *bi-gram*. Therefore, we decided to also used another smoothing technique: Laplace. This is a very simple but efficient function widely used when processing text. The formulation is as following:

$$p'_d(x) = \frac{\mathrm{f}_{x,d} + \alpha}{|d| + \alpha|v|} \quad \alpha > 0$$

Where $|v|$ is the number of different features, or in this case, the size of the vocabulary.

In the pre-validation, we noticed that the *profile-based approach* and the *instance-based approach* for this problem had different results depending on the collection. Therefore, we add these variants to our testing.

We have to mention that the pretesting or parameter adjustment was made using only the validation dataset in order to not alter the final testing.

3.3.3 Advantages

As mentioned before, according with the studies [42, 43, 44, 35], KLD is a strong option among the authorship attribution methods, we can mention some reasons:

Simplicity KLD is a simple approach for authorship attribution, its fundaments are theoretically simples based on entropy.

Effectiveness The previous studies demonstrate that it can work at least as well as SVM, with the advantage that it can work in multi-class problems and SVM does not.

Efficiency Comparing to SVM, KLD requires less resources to classify, even in cases of large data collections.

Manage large collections Comparing with other methods of Information Retrieval for large corpora, KLD demonstrates being fast and clearly identify of the author in the top-rank. It is a good option for authorship search.

3.3.4 Implementation Tools

The pre-processing part was developed in Java program language due to the large number of libraries to facilitate the management of files in different formats. The program environment was Netbeans IDE 7.2 with the language version Java 1.7.0_51 and the additional jar libraries JDK 1.7 and Jsoup 1.7.3. The processing was implemented in Python because it is a simple and fast language and it has a powerfull set of libraries for NLP and Machine Learning. We used Python 2.7.4 with libraries NLTK 2.7 and SciKit Learn 0.14.

Chapter 4

Evaluation and Analysis of Results

Contents

4.1 Pre-Analysis of Data

The first step is to understand and be familiar with the data. For this purpose, we perform a previous analysis of features. We are managing two main features: *function words* and *POS* with two corpora: *Spanish literature* and *blogs*, this last one has documents in English and Spanish.

About the corpora of Spanish literature, we can see in Figure 4.1 the most frequent words over all the books; and as expected, they generally coincide with the more frequent words in Spanish that are also *function words*. Then, in Figure 4.2 we can see the frequencies filtering all words but *function words* and their distribution over the authors. In this case, as well as before, the more frequent *function words* are almost similarly distributed among the authors because their use is extended. However, there are some initial clues to discriminate one author from another. For example *VICE* seems to use more frequently the word *"los"* than the others. Therefore, we use an external tool to analyse more in detail which are the words that can discriminate better. The tool is Weka [39] and we applied Information Gain evaluation, the results are in the Figure 4.3.

Ranking	Word	Frequency
1	,	392008
2	*de*	268839
3	*la*	175028
4	*que*	161772
5	*y*	161107
6	*el*	119638
7	*en*	112096
8	*a*	80079
9	*los*	72277
10	--	65650
11	*no*	64462
12	*se*	63508
13	*con*	62632
14	*su*	58091
15	*un*	52211
16	*las*	51945
17	*por*	48864
18	...	41795
19	*del*	39863
20	*una*	38053

Figure 4.1: Ranking of the most frequent words in *Spanish literature* corpora

After the Information Gained evaluation, we plot a table with the resulting top words (Figure 4.4); and highlight the differences. The values in red are unusual higher respect to the mean while the blue ones are lower. For example the presence of the word *"junto"* indicates that the document could be from *VICE* or *BENI* while its absence indicates that probably that document is from *FERN* or *PEDR*. The same case with the evaluation by gender, the words *"nuestras"* and *"enseguida"* look more related with males authors, while *"despacio"*, *"tuyas"* and *"ésas"* with females. This result is a little interesting because *"enseguida"* means *promptly* and it is related to males and *"despacio"* means *slowly*, related to females. Following the same idea we analysed the blogs in English, in Figure 4.6. We can see that in this sample the words *"the"* , *"and"* and *"of"* seems to indicate that the document was written by an older person while *"a"*, *"for"* and *"is"* by a younger one.

	de	*la*	*que*	*y*	*el*	*en*	*a*	*los*	*no*	*se*	*con*	*su*
Frequency	268839	175028	161772	161107	119638	112096	80079	72277	64462	63508	62632	58091
JOSE	0.16	0.09	0.10	0.11	0.06	0.07	0.05	0.04	0.04	0.03	0.03	0.03
EMIL	0.15	0.11	0.08	0.10	0.08	0.06	0.07	0.04	0.04	0.04	0.03	0.02
PEDR	0.15	0.10	0.10	0.10	0.08	0.06	0.09	0.03	0.03	0.03	0.03	0.02
JUAN	0.15	0.09	0.10	0.14	0.06	0.07	0.07	0.03	0.04	0.04	0.03	0.03
VICE	0.17	0.10	0.08	0.07	0.08	0.07	0.03	0.06	0.02	0.03	0.04	0.04
FERN	0.14	0.10	0.11	0.09	0.07	0.06	0.07	0.03	0.04	0.04	0.03	0.03
LEOP	0.15	0.09	0.09	0.09	0.06	0.06	0.07	0.03	0.04	0.04	0.03	0.03
CONC	0.14	0.12	0.06	0.11	0.07	0.07	0.06	0.03	0.03	0.04	0.04	0.03
BENI	0.15	0.10	0.10	0.10	0.06	0.06	0.06	0.03	0.05	0.04	0.03	0.03
ARMA	0.15	0.10	0.10	0.09	0.07	0.07	0.03	0.04	0.04	0.04	0.04	0.03

Figure 4.2: The most frequent *function words* and their relative frequency by author

Ranked	**Funtion Word**
0.978	*junto*
0.831	*tras*
0.799	*ante*
0.782	*casi*
0.683	*cuanto*
0.641	*de*
0.618	*todo*
0.584	*las*
0.566	*delante*
0.562	*muy*
0.541	*bajo*
0.515	*apenas*

Figure 4.3: Ranking of *function words* using an Information Gained evaluation (using Weka)

	junto	***tras***	***ante***	***casi***	***cuanto***	***las***	***delante***	***muy***	***bajo***	***apenas***
JOSE	2.12	0.94	1.50	2.25	7.99	77.35	2.00	16.92	3.68	1.50
EMIL	0.62	2.27	2.49	4.36	3.57	97.95	2.66	15.98	1.98	2.61
PEDR	0.15	0.00	1.07	2.61	4.29	69.45	1.69	22.38	3.37	0.31
JUAN	0.34	0.04	0.85	8.70	4.49	69.41	0.99	20.14	2.15	4.08
VICE	3.54	2.74	6.18	4.11	1.72	103.00	0.54	4.68	5.98	3.11
FERN	0.00	0.00	1.42	2.83	4.25	82.15	6.37	11.33	3.54	2.83
LEOP	1.78	0.52	2.18	6.01	5.25	82.10	2.15	18.38	2.02	1.35
CONC	1.00	1.50	2.24	4.99	2.24	70.54	3.49	28.91	4.74	4.24
BENI	2.36	1.79	2.09	3.22	2.97	85.93	2.47	23.03	1.86	1.58
ARMA	0.22	0.35	1.49	4.01	4.38	82.19	4.41	12.00	2.36	2.35

Figure 4.4: *Function words* with higher Information Gained by author. The red numbers are the cases when the value is unusually high respect to the mean, and the blue ones when the value is particular low.

	despacio	*nuestras*	*tuyas*	*ésas*	*ésa*	*esas*	*entre*	*enseguida*	*esa*	*es*
M	1.68	11.86	0.40	0.40	2.40	28.22	161.76	2.86	55.18	531.72
F	12.75	2.01	1.34	1.34	2.34	26.85	205.37	0.67	55.70	571.14

Figure 4.5: *Function words* with higher Information Gained by gender. The red numbers are the cases when the value is unusually high respect to the mean.

	the	*and*	*of*	*in*	*to*	*a*	*for*	*is*	*on*	*that*
50-64	0.249	0.111	0.118	0.081	0.115	0.092	0.047	0.052	0.031	0.047
65-xx	0.217	0.128	0.115	0.072	0.131	0.099	0.040	0.051	0.033	0.057
35-49	0.220	0.118	0.113	0.085	0.123	0.098	0.041	0.063	0.029	0.052
25-34	0.233	0.108	0.109	0.078	0.116	0.098	0.046	0.065	0.031	0.045
18-24	0.246	0.111	0.103	0.073	0.110	0.114	0.049	0.051	0.035	0.036

Figure 4.6: *Function words* with higher Information Gained by age in English blogs. The red numbers are the cases when the value in unusually high respect to the mean, and the blue ones when the value is particular low.

4.2 Evaluation and Analysis

In this section we explain the performed experiments. As mentioned before, we have as baseline the Naïve Bayes classifier and our solution is based on KLD classifier (with Dirichlet smoothing function). Some variants were added to the solution. First, we try two approaches: *profiled-based* and *instance-based*. Second, we test another smoothing function: Laplace, in order to compare and test the influence of an external model (Dirichlet use a background probability model). Finally, we try the variants of *uni-gram* and *bi-gram* for the POS tagger, however, the *bi-gram* is only applied to KLD with Laplace because at the moment we do not have a background model for bi-grams.

The evaluation consist on performing a 10-fold cross-validation using the accuracy as measure, and then, evaluating those results with a t-test to know if there is a significant difference between one variant or another. Finally, we test the solution with the testing dataset and analyse the results.

4.2.1 Spanish Literature from Project Gutenberg

The corpora of Spanish literature was divided in two different ways. One with partitions of 10000 lines and the other with 3000 lines (18 words per line approx.). In this case, we have two problems: recognizing the author's name and gender.

4.2.1.1 Authorship Attribution, Identifying the Author's Name

In Figures 4.7 and 4.8 we can see the results of the 10-fold cross-validation using the 10000-lines corpora with the baseline and all variants of our solution. We can see that there is difference between the naïve Bayes classifier (accuracy rate 30%) and the KLD classifier (accuracy rate 81%),. We can say that the solution is at least well enough for the problem. As variants, we can say that the *profiled-based approach* had better results, this is what we expect. In this case we can observe that Dirichlet and Laplace smoothing, in some cases, obtain the same results, this could be due to the parameter's setting. The parameter μ is 100 in our case because it was the value with better performance in the previous parameter

adjustment with validation data, we try values from 10e-4 to 10e+4 with a separation of one decimal. In [43] this parameter was set to $1000\sqrt{3}$. This parameter balance the weight of the model and the background model, therefore, we can say that the background model in this case is not determining the classification.

Spanish Literature (Gutenberg) 10000 words files

	Function Words				
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	14%	71%	86%	71%	100%
2	29%	43%	57%	43%	57%
3	57%	57%	71%	57%	57%
4	29%	100%	100%	100%	100%
5	43%	100%	100%	86%	100%
6	43%	86%	71%	86%	71%
7	29%	86%	86%	86%	86%
8	43%	100%	100%	100%	100%
9	0%	57%	57%	43%	29%
10	15%	92%	85%	92%	85%
Mean	**30%**	**79%**	**81%**	**76%**	**78%**

Figure 4.7: Accuracy rate in 10-folds for *Spanish literature 10000 lines* and function words by author

Spanish Literature (Gutenberg) 10000 words files

	POS					POS Bi-gram	
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	0%	86%	86%	71%	86%	100%	86%
2	0%	57%	71%	57%	71%	86%	71%
3	0%	71%	57%	71%	57%	86%	57%
4	14%	100%	100%	100%	100%	86%	100%
5	14%	86%	86%	86%	86%	71%	86%
6	14%	71%	86%	71%	86%	71%	86%
7	14%	100%	100%	100%	100%	86%	100%
8	14%	86%	86%	86%	86%	71%	86%
9	0%	29%	71%	29%	71%	29%	71%
10	0%	77%	85%	62%	85%	85%	85%
Mean	**7%**	**76%**	**83%**	**73%**	**83%**	**77%**	**83%**

Figure 4.8: Accuracy rate in 10-folds for *Spanish literature 10000 lines* and POS by author

In Figures 4.9 and 4.10, we did the same test for the files based in 3000 lines. In this case surprisingly, the *instance-based approach* obtained better results when using function words. We applied a t-test (two tailed with 5% of significance level) to each pair of variants. The results show not significant difference among all variants of KLD but only in the case of using a larger text per document (10000 lines) a *profile-based approach* is better and by using a smaller text per document (3000 lines) an *instance-based approach* seems better. This last observation could be due to in the *instance-based approach*, the models to calculate the distance have the same length, therefore, there is not bias for the cumulative quantity of features. The naïve Bayes classifier had low accuracy respect to KLD. This may be due to the smaller number of samples in the training because when using 3000 lines files (more samples) has better performance than when using 10000 lines files.

Spanish Literature (Gutenberg) 3000 words files

	Function Words				
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	65%	90%	80%	90%	85%
2	65%	70%	85%	65%	85%
3	55%	75%	80%	75%	80%
4	55%	80%	65%	75%	60%
5	70%	90%	55%	90%	50%
6	80%	90%	60%	90%	50%
7	85%	80%	100%	80%	100%
8	70%	70%	65%	75%	65%
9	65%	75%	60%	75%	60%
10	29%	90%	81%	90%	76%
Mean	**64%**	**81%**	**73%**	**81%**	**71%**

Figure 4.9: Accuracy rate in 10-folds for *Spanish literature 3000 lines* and function words by author

Spanish Literature (Gutenberg) 3000 words files

	POS					POS Bi-gram	
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	40%	90%	90%	90%	80%	90%	80%
2	45%	85%	85%	85%	85%	85%	85%
3	30%	80%	80%	75%	85%	85%	85%
4	45%	80%	80%	80%	60%	80%	60%
5	35%	80%	80%	75%	85%	70%	85%
6	50%	85%	85%	80%	90%	85%	90%
7	70%	85%	85%	90%	100%	90%	100%
8	60%	85%	85%	85%	90%	80%	90%
9	60%	85%	85%	80%	85%	75%	70%
10	10%	76%	76%	76%	71%	76%	71%
Mean	**44%**	**83%**	**83%**	**82%**	**83%**	**82%**	**82%**

Figure 4.10: Accuracy rate in 10-folds for *Spanish literature 3000 lines* and POS by author

The cross-validation was made with the overall training data. There were not significative difference between the text size, except in the case of the *instance-based approach*, which is different in 3000 lines files. The processing of 3000 lines files is faster. Therefore, we will continue with the 3000 lines files.

The testing was made with data separate from training and validation sets. No book parts where mixed, each book correspond to training, testing or validation sets in order to maintain the good practices. The Figures 4.11, 4.12 and 4.13 display the testing results. We can see the accuracy obtained for each author and the total one. The classifiers have problems to identify to the author *CONC*. This may be because this author only has two books in the collection; we can see it in more detail in the Figure 4.12 which is a sample of the classification log using KLD *profiled-based* with Dirichlet smoothing and *POS*. In this case, we can see that *CONC* has only three documents for training and the classifier fails to identify the two documents for the testing. Here, we also can notice that KLD with Dirichlet smoothing could classify more documents of *BENI* than the others. *BENI* is a sample having a larger quantity of samples to train and to test. The combination of *POS tags and words* improves significative the results of all classifiers, they obtained until 95% of total accuracy.

Spanish Literature (Gutenberg) 3000 words files

	Function Words				
	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
ARMA	75%	88%	62%	88%	62%
BENI	67%	92%	58%	92%	58%
CONC	0%	50%	100%	50%	100%
EMIL	50%	50%	100%	100%	100%
FERN	0%	100%	100%	100%	100%
JOSE	60%	80%	80%	80%	80%
JUAN	100%	100%	100%	100%	100%
LEOP	67%	100%	100%	100%	100%
PEDR	0%	0%	100%	0%	100%
VICE	80%	93%	93%	93%	93%
TOTAL	50%	88%	83%	90%	83%

Figure 4.11: Testing accuracy for *Spanish literature 3000 lines* and function words by author

Spanish Literature (Gutenberg) 3000 words files

	POS					POS Bi-gram	
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
ARMA	38%	100%	100%	100%	100%	100%	88%
BENI	25%	83%	83%	75%	58%	75%	58%
CONC	100%	0%	0%	0%	0%	0%	0%
EMIL	50%	50%	50%	50%	50%	50%	50%
FERN	0%	100%	100%	100%	100%	100%	100%
JOSE	80%	100%	100%	100%	80%	100%	80%
JUAN	50%	100%	100%	100%	100%	100%	100%
LEOP	67%	100%	100%	100%	100%	100%	100%
PEDR	0%	100%	100%	100%	100%	100%	100%
VICE	60%	93%	93%	93%	93%	93%	93%
Mean	47%	90%	90%	88%	83%	88%	81%

Figure 4.12: Testing accuracy for *Spanish literature 3000 lines* and POS by author

Spanish Literature (Gutenberg) 3000 words files

	POS and Words				
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
ARMA	88%	100%	100%	100%	100%
BENI	100%	92%	83%	92%	83%
CONC	0%	100%	0%	100%	0%
EMIL	0%	50%	0%	50%	0%
FERN	0%	100%	0%	100%	0%
JOSE	0%	100%	100%	100%	80%
JUAN	62%	100%	100%	100%	100%
LEOP	0%	100%	100%	100%	100%
PEDR	0%	100%	0%	100%	0%
VICE	80%	93%	93%	93%	93%
Mean	62%	95%	83%	95%	81%

Figure 4.13: Testing accuracy for *Spanish literature 3000 lines* combining POS and words by author

```
Accuracy: 0.897
Precision: 0.872
              accuracy precision   support  training

       ARMA       1.00      1.00         8        45
       BENI       0.83      1.00        12        40
       CONC       0.00      0.00         2         3
       EMIL       0.50      1.00         2         8
       FERN       1.00      1.00         2         2
       JOSE       1.00      1.00         5        11
       JUAN       1.00      1.00         8        16
       LEOP       1.00      1.00         3        11
       PEDR       1.00      1.00         1         9
       VICE       0.93      1.00        15        56

avg / total       0.90      0.97        58       201
```

Figure 4.14: Log sample from KLD, Dirichlet, profiled-based for *Spanish literature 3000 lines* and POS by author

4.2.1.2 Authorship Profiling, Identifying the Author's Gender

The second problem it to identify the author's gender. The data in this case is not equally distributed. We have 144 samples from males authors and 15 for females. Nevertheless, the classifiers can discriminated with a high accuracy between both options. In Figures 4.15 and 4.16, we made the same analysis. We can notice that in all features types: *function word*, *POS (uni-gram and bi-gram)* and *POS with words*, the better results are obtaining by using the *instance-based approach*. The reason could be the inequality of the distribution of samples. The *profile-based approach* accumulates more text to train for males than for females. By the contrary, in an *instance-based approach* the size of text for training is similar in both classes. In addition, we can notice that the feature *POS bi-gram* has the same performance that *POS*, thus, we can discard it because it takes more processing time, the combination of *POS tags and words* shows some improvement respect to the other features. Finally, the t-test results shows not significative difference by using one or other smoothing with this sample.

Spanish Literature (Gutenberg) 3000 lines files

	Function Words				
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	75%	100%	50%	95%	50%
2	55%	95%	100%	95%	100%
3	70%	100%	95%	100%	85%
4	85%	95%	70%	95%	70%
5	80%	95%	80%	95%	80%
6	80%	95%	75%	95%	75%
7	85%	100%	95%	100%	95%
8	80%	100%	100%	100%	100%
9	75%	100%	80%	100%	80%
10	52%	95%	57%	95%	57%
Mean	**74%**	**98%**	**80%**	**97%**	**79%**

Figure 4.15: Accuracy rate in 10-folds for *Spanish literature 3000 lines* and function words by gender

Spanish Literature (Gutenberg) 3000 lines files

	POS					POS Bi-gram	
K-fold	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	10%	100%	65%	100%	65%	100%	65%
2	25%	95%	95%	100%	95%	100%	95%
3	15%	100%	95%	100%	95%	100%	95%
4	40%	95%	70%	95%	75%	95%	75%
5	25%	95%	80%	95%	80%	95%	80%
6	10%	100%	100%	100%	100%	100%	100%
7	10%	100%	95%	100%	95%	100%	95%
8	20%	100%	100%	100%	100%	100%	100%
9	25%	100%	90%	100%	90%	100%	90%
10	14%	95%	71%	95%	71%	95%	71%
Mean	**19%**	**98%**	**86%**	**99%**	**87%**	**99%**	**87%**

Figure 4.16: Accuracy rate in 10-folds for *Spanish literature 3000 lines* and POS by gender]

The results of the testing are displayed in Figure 4.18. In general the classifiers can recognise the documents writing by males with better accuracy than by females. This is due the quantity of samples: large for males and small for females. In the test log

Spanish Literature (Gutenberg) 3000 lines files				
	POS and Words			
K-fold	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
1	100%	85%	95%	85%
2	95%	85%	100%	90%
3	100%	85%	100%	85%
4	95%	80%	100%	85%
5	95%	95%	95%	95%
6	100%	100%	100%	100%
7	100%	100%	100%	100%
8	100%	100%	100%	100%
9	100%	100%	100%	100%
10	95%	90%	95%	90%
Mean	98%	92%	99%	93%

Figure 4.17: Accuracy rate in 10-folds for *Spanish literature 3000 lines* combining POS and Words by gender

using KLD with Dirichlet smoothing with an *instance-based approach* and a combination of *POS with words* (Figure 4.19) we can see the distribution of samples: for males 190 training samples and 54 testing samples; while for females 11 training samples and 4 testing samples. We presume that with more sample data the accuracy will improve.

4.2.2 Blogs from PAN 2014

The second collection on data are *blogs* from PAN 2014, in English and Spanish. In this case we have two problems: identify the author's gender and range of age. Due to the large quantity of data managed in this case, we will only display the testing results

4.2.2.1 Authorship Profiling, Identifying the Author's Gender

The Figure 4.20 shows the results of applying the different methods over the blogs in English. The general accuracy is high, being the better cases the *instance-based approaches*. Again we think that the reason is the text size, as observed before, this approach works better when we manage short text (3000 lines or less according our observations). The best case was by using KLD with Dirichlet smoothing function with 100% accuracy, nevertheless, in the cross-validation was a variation from 85% to 100%. The highest values were obtained in the cases with more samples for training, in the Figure 4.21 we can see that the distribution of samples between females and males was similar, therefore, the accuracy do not vary by gender.

In the case of blogs in Spanish (Figure 4.22) the results were very different than for English. The accuracy for the methods based on KLD were low, not even better than a random solution, and the best results were obtained by the naïve Bayes classifier. The only reason that we find is the low quantity of text in the samples, at difference of the other collections, here we only have 80 documents of around 50 words each one (the English corpora has 255 documents). We can deduce that, naïve Bayes needs more samples to train. In order to improve, KLD needs a longer text size to capture the divergence with more accuracy. We have to mention that KLD can be improved with

Spanish Literature (Gutenberg) 3000 lines files

	Function Words				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	50.00%	75.00%	74.00%	100.00%	100.00%
M	63.00%	96.00%	76.00%	96.00%	76.00%
Mean	**62.10%**	**94.80%**	**75.90%**	**96.60%**	**77.60%**

	POS				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	100.00%	25.00%	75.00%	25.00%	75.00%
M	15.00%	100.00%	91.00%	100.00%	93.00%
Mean	**20.70%**	**94.80%**	**89.70%**	**94.80%**	**91.40%**

	POS and Words				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	0.00%	75.00%	0.96%	75.00%	0.00%
M	100.00%	100.00%	0.90%	100.00%	96.00%
Mean	**93.10%**	**98.30%**	**89.70%**	**98.30%**	**89.70%**

Figure 4.18: Testing accuracy for *Spanish literature 3000 lines* by gender

```
Accuracy: 0.983
Precision: 0.000
              accuracy precision   support  training

           F       0.75      0.00         4        11
           M       1.00      1.00        54       190

avg / total        0.98      0.93        58       201
```

Figure 4.19: Log sample from KLD, Dirichlet, instance-based for *Spanish literature 3000 lines* combining POS and words by gender

Blogs in English (PAN 2014)

	Function Words				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	100.00%	100.00%	64.00%	75.00%	56.00%
M	48.00%	100.00%	72.00%	96.00%	72.00%
Mean	**74.30%**	**100.00%**	**68.00%**	**85.30%**	**64.20%**

	POS				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	100.00%	100.00%	69.00%	84.00%	82.00%
M	70.00%	100.00%	61.00%	94.00%	39.00%
Mean	**85.30%**	**100.00%**	**65.00%**	**89.00%**	**60.60%**

	POS and Words				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	100.00%	100.00%	96.00%	82.00%	95.00%
M	43.00%	100.00%	93.00%	100.00%	85.00%
Mean	**71.60%**	**100.00%**	**94.00%**	**90.80%**	**89.90%**

Figure 4.20: Testing accuracy for *blogs in English* by gender

```
Accuracy: 1.000
Precision: 0.000
              accuracy precision   support  training

           F       1.00      0.00        55        73
           M       1.00      1.00        54        74

avg / total       1.00      0.50       109       147
```

Figure 4.21: Log sample from KLD, Dirichlet, instance-based for *blogs in English* and POS by gender

a adjustment of parameter. As mentioned before, we are using μ 100. The background model was extracted from IULA corpora based on articles from different topics with a scientific or technical orientation. Thus, it may not represent the ideal language usage and more tests are needed in this direction.

Blogs in Spanish (PAN 2014)

	Function Words				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	73%	55%	45%	55%	45%
M	71%	57%	57%	57%	57%
Mean	**72%**	**55%**	**50%**	**56%**	**50%**

	POS				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	82%	27%	45%	73%	36%
M	71%	43%	57%	0%	57%
Mean	**78%**	**33%**	**50%**	**44%**	**44%**

	POS and Words				
Gender	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
F	73%	91%	27%	73%	36%
M	71%	14%	71%	0%	57%
Mean	**72%**	**61%**	**44%**	**44%**	**44%**

Figure 4.22: Testing accuracy for *blogs in Spanish* by gender

```
Accuracy: 0.333
Precision: 0.000
             accuracy precision   support  training

          F      0.27      0.00        11        29
          M      0.43      1.00         7        33

avg / total      0.33      0.39        18        62
```

Figure 4.23: Log sample from KLD, Dirichlet, instance-based for *blogs in English* and POS by gender

4.2.2.2 Authorship Profiling, Identifying the Author's Age Range

As the same with the gender case, the results for blogs in English had a high accuracy while for Spanish it was very low. We can see this results in Figures 4.24 and 4.25, the analysis is also the same as before. Here for example, we have one case with more samples to train (range age 35-49). As shown in Figure 4.26) the classification with KLD is then improved.

Blogs in English (PAN 2014)

	Function Words				
Age	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
18-24	67	100%	100	100	100
25-34	96	100%	40	77	40
35-49	42	100%	57	90	47
50-64	47%	100%	82%	88%	71%
65-xx	50%	100%	50%	100%	50%
Mean	**67%**	**100%**	**55%**	**84%**	**50%**

	POS				
Age	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
18-24	67%	100%	33%	100%	33%
25-34	100%	100%	15%	81%	9%
35-49	90%	100%	47%	88%	30%
50-64	59%	100%	59%	76%	76%
65-xx	50%	100%	50%	100%	50%
Mean	**88%**	**100%**	**35%**	**83%**	**28%**

	POS and Words				
Age	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
18-24	0%	100%	100%	100%	100%
25-34	100%	100%	98%	72%	91%
35-49	40%	100%	100%	95%	100%
50-64	41%	100%	100%	71%	100%
65-xx	50%	100%	100%	100%	100%
Mean	**65%**	**100%**	**99%**	**82%**	**96%**

Figure 4.24: Testing accuracy for *blogs in English* by age

Blogs in Spanish (PAN 2014)

	Function Words				
Age	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
18-24	0	0%	0%	0%	0%
25-34	0	0%	29%	0%	0%
35-49	83	83%	50%	100%	83%
50-64	0%	0%	0%	0%	0%
65-xx	0%	0%	0%	0%	0%
Mean	**28%**	**28%**	**39%**	**33%**	**33%**

	POS				
Age	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
18-24	0%	0%	0%	0%	0%
25-34	86%	0%	29%	0%	0%
35-49	0%	83%	50%	83%	83%
50-64	0%	0%	0%	0%	0%
65-xx	0%	0%	0%	0%	0%
Mean	**33%**	**28%**	**28%**	**28%**	**28%**

	POS and Words				
Age	Naive Bayes	KLD-Dirichlet Instance-Based	KLD-Dirichlet Profiled-Based	KLD-Laplace Instance-Based	KLD-Laplace Profile-Based
18-24	0%	0%	0%	0%	0%
25-34	1%	0%	0%	0%	0%
35-49	40%	100%	100%	83%	83%
50-64	41%	0%	0%	0%	0%
65-xx	50%	0%	50%	0%	0%
Mean	**65%**	**33%**	**39%**	**28%**	**28%**

Figure 4.25: Testing accuracy for *blogs in Spanish* by age

```
Accuracy: 0.333
Precision: 0.111
              accuracy precision   support  training

      1824        0.00      0.00         1         2
      2534        0.00      0.00         7        17
      3549        1.00      1.00         6        32
      5064        0.00      0.00         2         9
      65xx        0.00      0.00         2         2

avg / total       0.33      0.33        18        62
```

Figure 4.26: Log sample from KLD, Dirichlet, instance-based for *blogs in English* and function word by age

Conclusion

The authorship attribution (AA) is the issue to infer the author of a given text by analysing its stylometry. It has a long history but the modern statistics and the advances in other areas of computer science made possible to archive good accuracy rates. It is based on the fact that each person has a unique and distinctive writing style which is independent on the topic, genre, formalist level and even its own consensus; this is called its *fingerprint*.

There are a number of interesting applications of this science, for example resolving disputes in literature works, forensics, plagiarism detection and many others. Although the statistics results of the current methods show that authorship attribution can be highly reliable in its predictions under certain conditions, there are still some issues to resolve. One of them is to define scientifically what features compose the *fingerprint*. There is theoretical support in linguistics to notice differences in the usage of certain function words, vocabulary, grammatical structure, and semantics. The science of stylometry is still in development. It works well when there is available large size of samples of the author's writing previously to the prediction because the algorithms have enough data to learn and find discriminative patterns. Nevertheless, in the real world there is not always such data, thus, methods must improve. Interpreting the natural language correctly is another challenge, the current methods have still a considerable error margin which can derives into an incorrect prediction. Finally, it is important to notice that there are still work to do with other languages than English.

The AA process starts with a pre-processing of the raw text in order to obtain the relevant part and discard the rest. The difficult task is to extract the correct features representative of the author writing style and to model them. The attribution methods learn from these features and predict the possible author or some of its characteristics as gender or age. Finally an evaluation is needed to prove its validity.

The features that have demonstrated a better performance are *function words* and *part-of-screech (POS) tagging*. The functions words permit an analysis independent on the topic and content; the part-of-screech tags permit to recognise the distinctive grammatical constructions of each author. The learning methods that generally have obtained better results are Support Vector Machine (SVM) and Bayes-Based methods. However, there are many others with important results such as Kullback-Leibler Divergence (KLD).

The problem that concerns to this project is to identify the authors of two different collections of data: *Spanish literature* and *blogs*, this last one in Spanish and English. Additionally, we have a *closed class* problem because we have a list of possible authors a priory, and also a *profiling problem* because we want to identify their gender and age. We choose this collections to experiment the performance of one of the current methods using other language like Spanish and with other characteristics of the problem like gender and age which are topics that are recently being explored.

Our proposal is to use KLD because it works with different text genre, it is fast, efficient and effective, some studies demonstrated that it performs better than other methods

and at least as well as SVM in a *closed class* problem, and it can be implemented with function words and POS tagging, which are the ones that represent the author's writing style with more accuracy at the moment. The baseline to compare our solution was naïve Bayes classifier because it is easy to implement and works well in this kind of problem, therefore is a good reference method. We used the library of NLTK to implement it. We use a *bag-of-words* model to represent the language, it consist on interpret the text as a list of the words that compose it without taking into account the order and relations among them; we use a vector to managed the features. To extract the function words we just filter the list of total words using a given list of function words. For English, we used the list of 344 items that was proposed previously in the study of Zhao and Zobel and for Spanish, we used a set of 307 most frequent function words in Spanish.

To extract the POS tags we used the NLTK parser, this parser requires a previous tagged set of tagged sentences, thus, we used the corpora Brown for English and IULA for Spanish. The parser can process text in different *n-grams* (sequences of n words), we use uni-gram and bi-gram sequences (one and two consecutive words). We tested some variants using KLD, one *profiled-based approach* that cumulates all the text of a given author and creates a unique profile based on it, then it calculates the KLD of the profile and the questioned document. The other was a *instance-based approach*, which calculates the KLD for each pair of documents. The selection for both approaches is based in the nearest document or profile under the assumption that while shorter is the difference between two models it is most probably that both were written by the same person, or by a person with the same gender or age.

Another variant was the smoothing function, this function is important in the cases when the original probability is zero. It smooth the probability to avoid this problem. We use Dirichlet and Laplace smoothing, Dirichlet use a background model of word probabilities, that represent the general usage of the language. For English we extract the model using the Brown corpora and for Spanish the IULA corpora; this background is used to bias the model into the most frequent words. Not always is possible to have a good background model, we had this problem while creating a background model for the POS feature using bi-gram parser, that is why we tested the other smoothing function: Laplace, this function is simple, is commonly used in text processing and does not require external models.

We evaluated our solution using the accuracy (the quantity of well classified documents over the total). First, we perform a 10-fold cross-validation. Then, a test with a separate dataset from the training one to not bias the model. To compare the methods we used a two tailed t-test (with 95% of confidence level). The evaluation results shows that the KLD is a good alternative solution for authorship attribution using Spanish language when we have enough data for training. In the evaluation using Spanish literature, the KLD classifier had better results than the Naïve Bayes Classifier. The performance of the Naïve Bayes was poor because there were few samples for each author. The smoothing function seems to be not relevance in this case, we notice that the text size is relevant to choose between a profile-based or an instance-based approach. The profile-based seems to works better in the cases of larger text per document (10000 lines of 18 words per line approx.). The instance-based seems to work better with shorter or medium size text (3000 lines). The option with statistically significative better accuracy was KLD with Dirichlet smoothing. On the contrary when we classified the blogs in Spanish, the performance of KLD was lower than the Naïve Bayes, we think that it is because there were very few text to train by document. We have 80 samples documents for training with around 50 words each one. There was a particular case when it was 32 samples to train and the results

of KLD were much more better than with the Naïve Bayes classifier for that particular case. Our solution needs a minimum quantity of text to perform well, we estimate at least 1500 words per author, however, it could improved with a better background model. For Spanish, our model was extracted from IULA corpora, this corpora is based on scientific and technical articles in Spanish, therefore, it could be not representative of the general language usage; for future work it could be important to improve this model.

In the case of classifying the gender, the KLD had better results than the Naïve Bayes (with the exception of the blogs in Spanish); as we mentioned before it seems that the instance-based approach is better for short or medium size text, and the difference on using Dirichlet or Laplace seems not significant when we have large quantity of samples. One particular feature that showed better results to discriminated between males and females was the combination of POS and words; it obtained until 98% of accuracy in Spanish literature and 100% with blogs.

In the case of age range identification, we only had the blogs set to evaluate; in the English case the KLD with Dirichlet smoothing had a remarkable performance using any features in an instance-based approach, we can mention also that POS combining with words had better accuracy to predict than other features for all the methods; in the case of Spanish as we mentioned the samples were too few.

Finally we can conclude that KLD with Dirichlet smoothing is a good option when using enough quantity of text for training. It is needed to test a better background model to prove its performance with few samples. In general we can say that for documents with large text size, a profile-based approach works better and the background model seems to be not significant. On the contrary, with short or middle size text by document the instance-based approach is better and the background model seems to be significant. In the case of features, the one that seems to represent better the writting style of the authors is the combination of POS tags and words.

The value of this project was to test a method that worked well with English corpora and closed class problem, with a different language: Spanish; and different problem: profiling of gender and age. More work is needed but we hope have contribute a little to this science.

Bibliography

[1] Aarts, B., Chalker, S., and Weiner, E. *The Oxford Dictionary of English Grammar*. Opr Series. OUP Oxford, 2014.

[2] Argamon, S., and Levitan, S. Measuring the usefulness of function words for authorship attribution. In *ACH/ALLC* (2005).

[3] Argamon, S., Whitelaw, C., Chase, P., Hota, S. R., Garg, N., and Levitan, S. Stylistic text classification using functional lexical features. *Journal of the American Society for Information Science and Technology 58*, 6 (2007), 802–822.

[4] Bird, S. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions* (2006), Association for Computational Linguistics, pp. 69–72.

[5] Bird, S., Klein, E., and Loper, E. *Natural language processing with Python.* " O'Reilly Media, Inc.", 2009.

[6] Burrows, J. 'an ocean where each kind. . .': Statistical analysis and some major determinants of literary style. *Computers and the Humanities 23*, 4-5 (1989), 309–321.

[7] Burrows, J. F. Word-patterns and story-shapes: The statistical analysis of narrative style. *Literary and Linguistic Computing 2*, 2 (1987), 61–70.

[8] Chaski, C. E. Who's at the keyboard? authorship attribution in digital evidence investigations. *IJDE 4*, 1 (2005).

[9] Craig, D., and Kinney, A. *Shakespeare, Computers, and the Mystery of Authorship.* Cambridge University Press, 2009.

[10] de Marneffe, M.-C., and Manning, C. D. *Stanford Dependencies manual*, 3.3 in december 2013 ed. The Stanford Natural Language Processing Group, September 2008.

[11] for Text Mining (NaCTeM), T. N. C. Text mining resources corpora.

[12] Francis, W. N., and Kucera, H. Brown corpus manual. *Brown University Department of Linguistics* (1979).

[13] Grieve, J. Quantitative authorship attribution: An evaluation of techniques. *Literary and linguistic computing 22*, 3 (2007), 251–270.

[14] Group, T. S. N. L. P. Statistical natural language processing and corpus-based computational linguistics: An annotated list of resources.

[15] Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., and Tibshirani, R. *The elements of statistical learning*, vol. 2. Springer, 2009.

[16] Holmes, D. I. Authorship attribution. *Computers and the Humanities 28*, 2 (1994), 87–106.

[17] Holmes, D. I. The evolution of stylometry in humanities scholarship. *Literary and Linguistic Computing 13*, 3 (1998), 111–117.

[18] Holmes, D. I., and Forsyth, R. S. The federalist revisited: New directions in authorship attribution. *Literary and Linguistic Computing 10*, 2 (1995), 111–127.

[19] Jockers, M. L., and Witten, D. M. A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing* (2010), fqq001.

[20] Joyce, J. M. Kullback-leibler divergence. In *International Encyclopedia of Statistical Science*. Springer, 2011, pp. 720–722.

[21] Juola, P. Authorship attribution. *Found. Trends Inf. Retr. 1*, 3 (Dec. 2006), 233–334.

[22] Koppel, M., Schler, J., and Argamon, S. Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology 60*, 1 (2009), 9–26.

[23] Koppel, M., Schler, J., and Argamon, S. Computational methods in authorship attribution. *Journal of the American Society for information Science and Technology 60*, 1 (2009), 9–26.

[24] Lebert, M. Project gutenberg (1971-2008). Online, http://www.gutenberg.org/cache/epub/27045/pg27045.html, October 2008.

[25] Luyckx, K., and Daelemans, W. The effect of author set size and data size in authorship attribution. *Literary and Linguistic Computing 26*, 1 (2011), 35–55.

[26] Manning, C. D., and Schütze, H. *Foundations of statistical natural language processing*. MIT press, 1999.

[27] Marimon, M., Fisas, B., Bel, N., Villegas, M., Vivaldi, J., Torner, S., Lorente, M., Vázquez, S., and Villegas, M. The iula treebank. In *LREC* (2012), pp. 1920–1926.

[28] Mendenhall, T. C. The characteristic curves of composition. *Science 9*, 214 (1887), pp. 237–249.

[29] Mendenhall, T. C. A mechanical solution of a literary problem. *Popular Science 60* (1901), pp. 97–105.

[30] Mosteller, F., and Wallace, D. L. Inference in an authorship problem. *Journal of the American Statistical Association 58*, 302 (1963), pp. 275–309.

[31] Peng, F., and Schuurmans, D. Combining naive bayes and n-gram language models for text classification. In *Advances in Information Retrieval*, F. Sebastiani, Ed., vol. 2633 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2003, pp. 335–350.

[32] Peng, F., Schuurmans, D., and Wang, S. Augmenting naive bayes classifiers with statistical language models. *Information Retrieval 7*, 3-4 (2004), 317–345.

[33] Rangel, F., Rosso, P., Chugur, I., Potthast, M., Trenkmann, M., Stein, B., Verhoeven, B., and Daelemans, W. Overview of the 2nd author profiling task at pan 2014.

[34] Rudman, J. The state of authorship attribution studies: Some problems and solutions. *Computers and the Humanities 31*, 4 (1997), 351–365.

[35] Savoy, J. The federalist papers revisited: A collaborative attribution scheme. In *Proceedings of the 76th ASIS&T Annual Meeting: Beyond the Cloud: Rethinking Information Boundaries* (Silver Springs, MD, USA, 2013), ASIST '13, American Society for Information Science, pp. 27:1–27:8.

[36] Stamatatos, E. A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology 60*, 3 (2009), 538–556.

[37] Toutanova, K., Klein, D., Manning, C. D., and Singer, Y. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1* (2003), Association for Computational Linguistics, pp. 173–180.

[38] Weiss, S. M., Indurkhya, N., Zhang, T., and Damerau, F. *Text mining: predictive methods for analyzing unstructured information*. Springer, 2010.

[39] Witten, I. H., and Frank, E. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.

[40] Zhai, C., and Lafferty, J. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (2001), ACM, pp. 334–342.

[41] Zhao, Y., and Zobel, J. Effective and scalable authorship attribution using function words. In *Information Retrieval Technology*, G. Lee, A. Yamada, H. Meng, and S. Myaeng, Eds., vol. 3689 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2005, pp. 174–189.

[42] Zhao, Y., and Zobel, J. *Entropy-based authorship search in large document collections*. Springer, 2007.

[43] Zhao, Y., and Zobel, J. Searching with style: Authorship attribution in classic literature. In *Proceedings of the Thirtieth Australasian Conference on Computer Science - Volume 62* (Darlinghurst, Australia, Australia, 2007), ACSC '07, Australian Computer Society, Inc., pp. 59–68.

[44] Zhao, Y., Zobel, J., and Vines, P. Using relative entropy for authorship attribution. In *Information Retrieval Technology*. Springer, 2006, pp. 92–105.

[45] Zipf, G. *Selected Studies of the Principle of Relative Frequency in Language*. Harvard University Press, 1932.